迴圈 - while

程式碼不是只會被執行一次

- 我們已學過「循序」和「條件分支」兩種程式結構。
- 程式碼會被執行 1 或 0 次。
- 接下來我們要看的是「重覆」結構，程式碼會被執行多次。我們稱之為「迴圈(Loop)」

範例：在畫面中顯示 1~10

- 你可以接受這種做法嗎？

```
cout << 1 << endl;
cout << 2 << endl;
cout << 3 << endl;
cout << 4 << endl;
cout << 5 << endl;
cout << 6 << endl;
cout << 7 << endl;
cout << 8 << endl;
cout << 9 << endl;
cout << 10 << endl;
```

```
1
2
3
4
5
6
7
8
9
10
```

這樣呢？

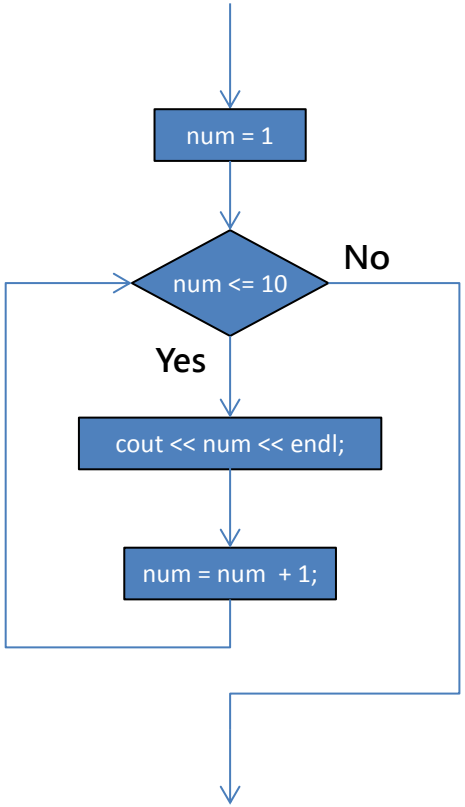
- 請修改剛剛那個程式，在畫面中顯示 1~10000。

更好的方法

- 使用 while 迴圈

```
int num = 1;

while(num<=10)
{
    cout << num << endl;
    num = num + 1;
}
```

```
1
2
3
4
5
6
7
8
9
10
```



while 的基本語法

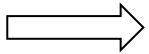
```
while (條件運算式)
{
      ......
      // 要重覆做的事情
      ......
}
```

- while 後面的條件運算式若成立(為 true)，則循序執行大括號內的程式。
- 待執行完大括號內的程式後，再次看條件運算式目前是否成立。
  - 若成立，再次循序執行大括號內的程式。
  - 若不成立，則離開 while 迴圈，繼續向下執行。

實作：使用 debug 工具觀察程式的執行

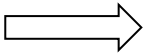
修改剛剛那個程式，在畫面中顯示 1~10000

```
int num = 1;

while(num<=10)
{
    cout << 1 << endl;
    num = num + 1;
}
```



修改剛剛那個程式，在畫面中顯示 1~10000

```
int num = 1;

while(num<=10)
{
    cout << 1 << endl;
    num = num + 1;
}
```



```
int num = 1;

while(num<=10000)
{
    cout << 1 << endl;
    num = num + 1;
}
```

這段程式有什麼問題？

```
int num = 1;

while(num<=10)
{
    cout << 1 << endl;
}
```

這段程式有什麼問題？

```
int num = 1;

while(num<=10)
{
    cout << 1 << endl;
}
```

- 有可能一不小心，就讓 while 迴圈變成一個「無窮迴圈」。
- 你可以動手試試看。
- 按下 Ctrl+C 可以中斷執行中的程式。

範例：輸出 n 的九九乘法表

```
int n = 0;

cout << "Please input n: ";
cin >> n;

int num=1;
while(num<10)
{
    cout << n << " x " << num << " = " << n*num << endl;
    num = num + 1;
}
```

```
Please input n: 3
3 x 1 = 3
3 x 2 = 6
3 x 3 = 9
3 x 4 = 12
3 x 5 = 15
3 x 6 = 18
3 x 7 = 21
3 x 8 = 24
3 x 9 = 27
```

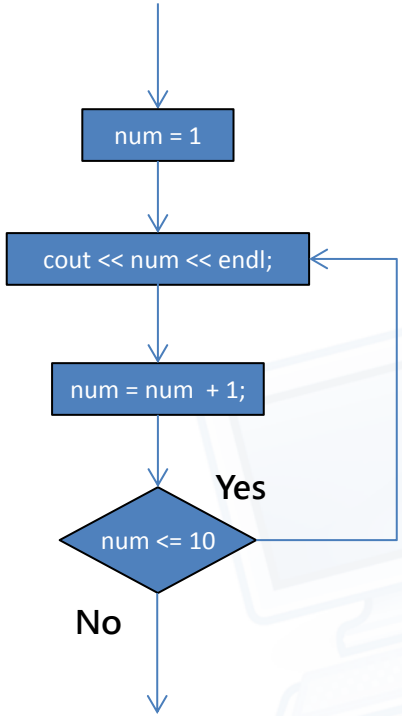
do...while

```
do
{
      ......
      //  要重覆做的事情
      ......
} while (條件運算式);
```

注意：這裡有分號

```
int num = 1;

do
{
    cout << num << endl;
    num = num + 1;
} while(num<=10);
```

```
1
2
3
4
5
6
7
8
9
10
```



比較 while 和 do...while...

- 一個先檢查條件是否成立再做事，一個先做事再檢查條件是否成立
- 一個可能連一次都不執行，一個至少執行一次

Break 可以跳出 while / do...while 迴圈

```
int id = 0;

while(true)
{
    cout << "請輸入顧客代碼(輸入 0 結束)：";
    cin >> id;

    if(id==0)
    {
        break;
    }
    else
    {
        // 進行該顧客相關作業
    }
}
```